

What is claimed is:

1 1. The method of responding to an incoming request message from a sender which comprises, in
2 combination, the steps of:

3 converting said incoming request message into an incoming canonical request message
4 expressed in a predetermined standard form,

5 comparing said incoming canonical request message with previously received and stored
6 canonical request messages, and

7 if a match is found between said incoming canonical request message and a given
8 previously stored canonical request message, accessing a stored response previously transmitted
9 in response to said given previously stored canonical message, and returning said stored response
10 to said sender.

11 2. The method of responding to an incoming request message as set forth in claim 1 wherein at
12 least a portion of said incoming request message is expressed in the Extensible Markup
13 Language and wherein said step of converting translates said portion into standard canonical
14 XML form.

1 3. The method of responding to an incoming request message as set forth in claim 1 wherein said
2 step of comparing comprises the substeps of:

3 generating an access key value based on the content of said incoming canonical request
4 message;

5 accessing zero or more selected ones of said previously received and stored canonical
6 request messages which are specified by said access key value, and

7 comparing said incoming canonical request message with said selected ones of said
8 previously received and stored canonical request messages.

1 4. The method of responding to an incoming request message as set forth in claim 3 wherein,
2 when no match is found between said incoming canonical request message and a previously

3 stored canonical request message, performing the step of storing said incoming canonical request
4 message in a first storage location specified by said access key.

1 5. The method of responding to an incoming request message as set forth in claim 4 wherein,
2 when no match is found between said incoming canonical request message and a previously
3 stored canonical request message, performing the further steps of:

4 generating a new response message containing data specified by said incoming request
5 message,
6 transmitting said new response message to said sender, and
7 storing said new response message at a second location associated with said first location.

4 6. The method of responding to an incoming request message expressed in the Extended Markup
2 Language which comprises, in combination, the steps of:

3 receiving said incoming request message via the Internet from a remote sender
4 converting said incoming request message into an incoming canonical request message
5 expressed in an established standard format,
6 comparing said incoming canonical request message with previously received and stored
7 canonical request messages,
8 if a match is found between said incoming canonical request message and a given
9 previously stored canonical request message, accessing a stored response previously transmitted
10 in response to said given previously stored canonical message, and returning said stored response
11 to said sender, and

12 if a match is not found between said incoming canonical request message and a given
13 previously stored canonical request message, performing the steps of:

14 generating a new response message containing data specified by said incoming
15 request message,
16 transmitting said new response message to said sender, and
17 storing said incoming canonical request message and said new response message
18 at associated storage locations.

1 7. The method of responding to an incoming request message as set forth in claim 6 wherein said
2 step of comparing comprises the substeps of:

3 generating an access key value based on the content of said incoming canonical request
4 message;

5 accessing zero or more selected ones of said previously received and stored canonical
6 request messages which are specified by said access key value, and

7 comparing said incoming canonical request message with said selected ones of said
8 previously receive and stored canonical request messages.

1 8. The method of caching XML request messages and the responses thereto transmitted via the
2 Internet which comprises, in combination, the steps of:

3 receiving an inbound HTTP message containing a request expressed in Extended Markup
4 Language from a sender,

5 translating said request into an inbound canonical request expressed into an inbound
6 canonical request expressed in a predetermined standard canonical format, and

7 comparing said inbound canonical request with previously stored canonical requests, and,
8 if a match is found with a particular one of said stored canonical requests, returning to said
9 sender a stored copy of a response message previously transmitted in response to said particular
10 one of said stored canonical requests.

1 9. The method of responding to an incoming request message as set forth in claim 8 wherein said
2 step of comparing comprises the substeps of:

3 generating an access key value based on the content of said inbound canonical request
4 message;

5 accessing zero or more selected ones of said previously received and stored canonical
6 request messages which are specified by said access key value, and

7 comparing said incoming canonical request message with said selected ones of said
8 previously receive and stored canonical request messages.

1 10. Apparatus for responding to an incoming request message which comprises, in combination,
2 means for receiving said request message from a remote sender via a data
3 communications link,
4 a translator for converting said incoming request message into an incoming canonical
5 request message expressed in a predetermined standard form,
6 a request cache memory for storing received canonical request messages,
7 a comparator for matching said incoming canonical request message with previously
8 received canonical request messages in said request cache memory,
9 a response cache memory,
10 means coupled to said comparator and responsive to a match between said incoming
11 canonical request message and a given previously stored canonical request message for
12 identifying a previously transmitted response to said given previously stored canonical message,
13 and
14 transmission means for sending said previously transmitted response to said remote
15 sender via said communications link..

1 11. The apparatus set forth in claim 10 wherein at least a portion of said incoming request
2 message is expressed in the Extensible Markup Language and wherein translator converts said
3 portion into standard canonical XML form.

1 12. The apparatus set forth in claim 10 wherein said comparator comprises:
2 means for generating an access key value based on the content of said incoming canonical
3 request message;
4 means for retrieving zero or more selected ones of said previously received and stored
5 canonical request messages from locations in said request cache memory which are specified by
6 said access key value, and
7 means for comparing said incoming canonical request message with said selected ones of
8 said previously received and stored canonical request messages

1 13. The apparatus set forth in claim 12 further including means responsive to the condition
2 occurring when no match is found between said incoming canonical request message and a
3 previously stored canonical request message for storing said incoming canonical request message
4 at a location in said request cache memory specified by said access key.

1 14. The apparatus set forth in claim 13 further wherein said means responsive to the condition
2 when no match is found between said incoming canonical request message and a previously
3 stored canonical request message further includes:

4 means for generating a new response message containing data specified by said incoming
5 request message,

6 means for transmitting said new response message to said sender, and

7 means for storing said new response message in said response cache memory.

8 15. Apparatus for responding to an incoming request message expressed in the Extended
9 Markup Language which comprises, in combination:

10 an Internet connection for receiving said incoming request message via the Internet from
11 a remote sender,

12 a translator for converting said incoming request message into an incoming canonical
13 request message expressed in an established standard format,

14 a cache memory for storing previously received and converted canonical request
15 messages and corresponding previously transmitted responses to said previously received request
16 messages,

a comparator for comparing said incoming canonical request message with said
previously received and stored canonical request messages in said cache memory,

means coupled to said comparator and responsive to a detected match between said
incoming canonical request message and a given previously stored canonical request message for
identifying that given previously transmitted response in said cache memory that was transmitted
in response to said given previously stored canonical request, and for transmitting said given
response to said remote sender via said Internet connection.

1 16. The apparatus set forth in claim 15 wherein said comparator comprises:
2 means for generating an access key value based on the content of said incoming canonical
3 request message;
4 means for retrieving zero or more selected ones of said previously received and stored
5 canonical request messages from locations in said cache memory which are specified by said
6 access key value, and
7 means for comparing said incoming canonical request message with said selected ones of
8 said previously received and stored canonical request messages.

1 17. In combination with a Web database server, a cache memory system for storing XML
2 request messages and the responses thereto, said cache memory system comprising, in
3 combination,
4 an Internet connection for receiving HTTP request messages from and returning HTTP
5 response messages to a remote client,
6 an inbound message port for receiving HTTP request messages at least some of which
7 contain a request payload expressed in Extended Markup Language,
8 a translator for converting each request payload into an inbound canonical request which
9 conforms to a predetermined standard canonical format,
10 a memory for storing previously received inbound canonical requests and the outbound
11 responses thereto,
12 a comparator for comparing each inbound canonical request canonical request with
13 previously stored canonical requests in said memory to identify a matching one of said stored
14 canonical requests, and
15 transmission means coupled to said comparator for returning a stored copy of that
16 previously transmitted response in said memory that was previously transmitted in response to
17 said matching one of said stored canonical requests.

1 18. The apparatus set forth in claim 17 wherein said comparator comprises:
2 means for generating an access key value based on the content of said inbound canonical
3 request message;

